

FIG. 1

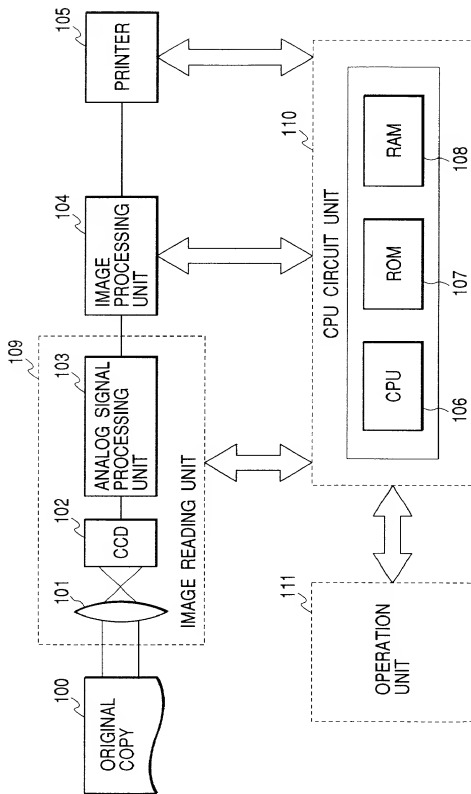


FIG. 2

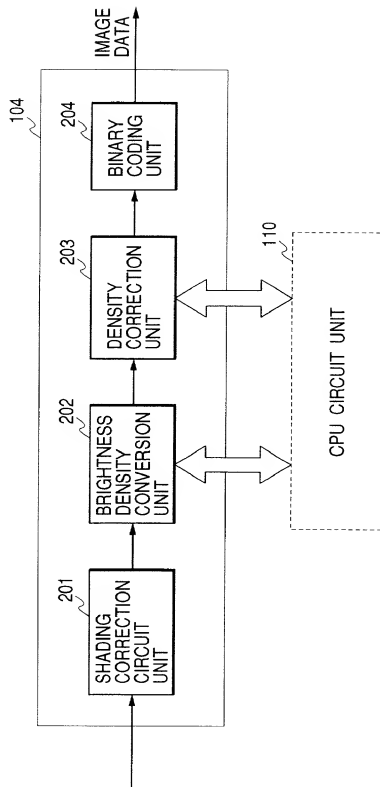


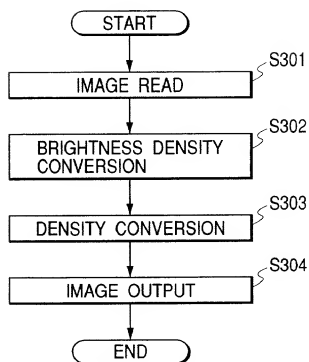
FIG. 3

FIG. 4

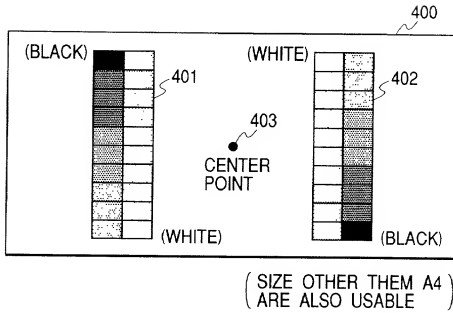


FIG. 5

96	80	64	48	32	16	12	8	4	0
255	240	224	208	192	176	160	144	128	112

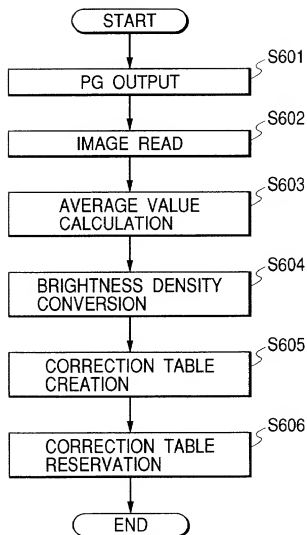
FIG. 6

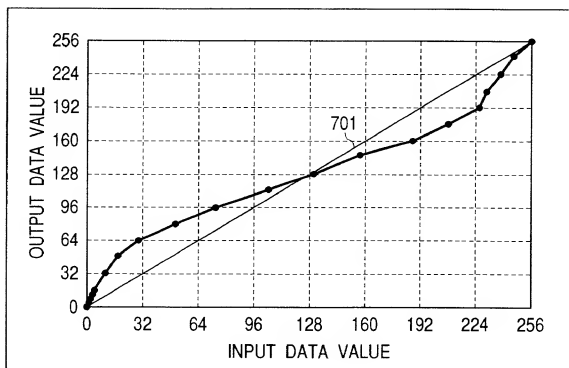
FIG. 7

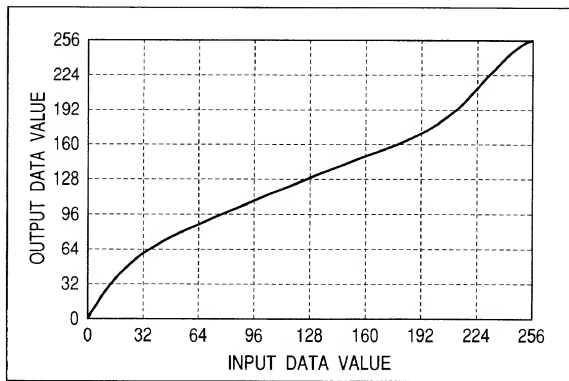
FIG. 8

FIG. 9

```

[ CORRECTION TABLE PREPARING UNIT (SMOOTHING) ]

( PROGRAM EXAMPLE 1 )

int i, j, sum ;
int buf [256] ;

for (i=1 ; i<255 ; i++) {
    sum=0 ;
    for (j=0 ; j<3 ; j++) {
        sum += density [i-1+j] ;
    }
    buf [i] = (sum/3) ;
}

/* BUFFER TO BE TEMPORALITY
   ENSURED */

/* RANGE OF j CORRESPONDS TO
   SMOOTHING WIDTH */

for (i=1 ; i<255 ; i++) {
    density [i] = buf [i] ;
}

/* FEEDBACK DATA AFTER
   SMOOTHING */

```


FIG. 10

```
[ CORRECTION TABLE PREPARING UNIT (SMOOTHING) ]
(PROGRAM EXAMPLE 2)

int i, j, sum ;
int buf [256] , buf2 [256] ;
for (k=0 ; k<3 ; k++) {
    for (i=1 ; i<254 ; i++) {
        sum=0 ;
        for (j=0 ; j<3 ; j++) {
            sum += buf [i-1+j] ;
        }
        buf2 [i] = (sum/3) ;
    }
    for (i=1 ; i<254 ; i++) {
        buf [i] = buf2 [i] ;
    }
}

/* BUFFER TO BE TEMPORALITY ENSURED */
/* RANGE OF k CORRESPONDS TO NUMBER
   OF TIMES FOR SMOOTHING */
/* RANGE OF j CORRESPONDS TO
   SMOOTHING WIDTH */

/* FEEDBACK DATA AFTER SMOOTHING */
```

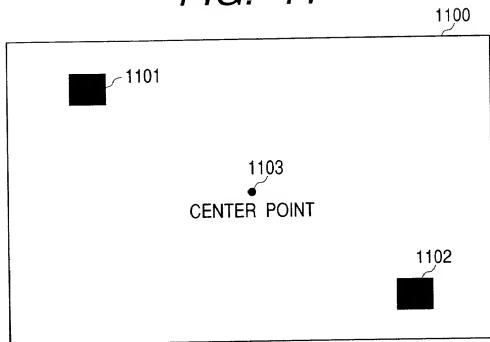
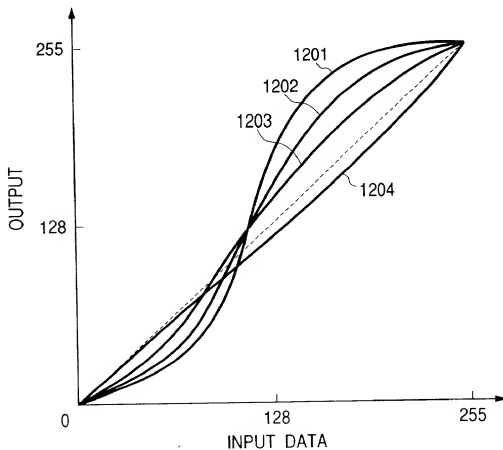
FIG. 11**FIG. 12**

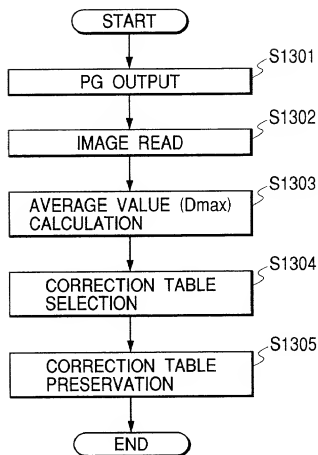
FIG. 13

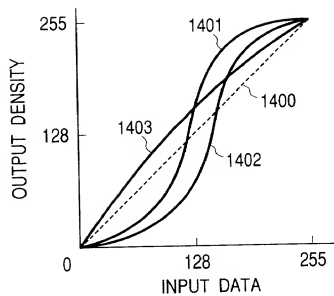
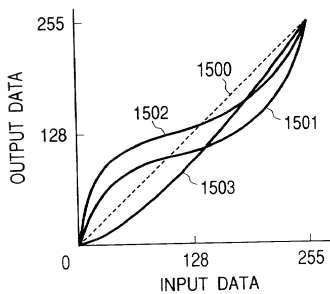
FIG. 14**FIG. 15**

FIG. 16

